Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.02 Осуществление интеграции программных модулей

*(наименование профессионального модуля)*

Выполнил:

обучающийся учебной группы № 519

В. О. Жиганов

*(И.О. Фамилия)*

Проверил:

руководитель практики от колледжа:

Т.Г. Аксёнова

*(И.О. Фамилия)*

М.В. Морозова

*(И.О. Фамилия)*

О.Г. Пташкин

*(И.О. Фамилия)*

И.В. Сибирев

*(И.О. Фамилия)*

**Москва**

**20­­­­23**

**Перечень заданий/работ, выполненных в ходе учебной практики**

Специальность 09.02.07 «Информационные системы и программирование»

*(наименование специальности)*

Профессиональный модуль ПМ.02 Осуществление интеграции программных модулей

*(наименование профессионального модуля)*

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Темы учебной практики** | **Выполненные задания/работы** |
| 1 | Тема 1. Использование выбранной системы контроля версий | Организация работы с проектом с использованием системы контроля версий Git. Создание и клонирование репозитория Git. Фиксация и извлечение изменений в проекте, отправка коммитов, работа с ветвями |
| 2 | Тема 2. Использование методов для получения кода с заданной функциональностью и степенью качества | Реализация методов для получения кода с заданной функциональностью и степенью качества |
| 3 | Тема 3. Интеграция модулей в программное обеспечение | Осуществление интеграции программных модулей в программное обеспечение |
| 4 | Тема 4. Отладка программных модулей | Выполнение отладки программных модулей с использованием специализированных программных средств |
| 5 | Тема 5. Разработка тестовых наборов и тестовых сценариев для программного обеспечения | Осуществление разработки тестовых наборов и тестовых сценариев для программного обеспечения |
| 6 | Тема 6. Инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования | Проведение инспектирования компонент программного обеспечения на предмет соответствия стандартам кодирования |
| 7 | Тема 7. Разработка технического задания на веб-приложение в соответствии с требованиями заказчика | Разработка и оформление технического задания на веб-приложение в соответствии с требованиями заказчика |
| 8 | Тема 8. Разработка веб-приложения в соответствии с техническим заданием | Разработка веб-приложения в соответствии с техническим заданием |
| 9 | Тема 9. Разработка интерфейса пользователя веб-приложения в соответствии с техническим заданием | Разработка интерфейса пользователя веб-приложения в соответствии с техническим заданием |
| 10 | Тема 10. Осуществление технического сопровождения и восстановления веб-приложения в соответствии с техническим заданием | Осуществление технического сопровождения и восстановления веб-приложения в соответствии с техническим заданием |

|  |  |  |
| --- | --- | --- |
| 11 | Тема 11. Тестирование разработанного веб-приложения | Осуществление тестирования разработанного веб-приложения |
| 12 | Тема 12. Размещение веб-приложения в сети в соответствии с техническим заданием | Размещение веб-приложения в сети в соответствии с техническим заданием |

**Содержание**

1. [**Тема: «ТЦ, Павильоны»** 5](#_Toc129821101)
2. [**Тема: «Товары автосервиса»** 45](#_Toc129821102)
3. [**Тема: «ГИБДД»** 66](#_Toc129821103)
4. [**Тема: «Дизайн на HTML и CSS»** 96](#_Toc129821104)
5. [**Тема: «HTML и CSS на Jupiter»** 110](#_Toc129821105)

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ №1**

# **Тема: «ТЦ, Павильоны»**

Студенты: Жиганов В. О.

Группа: 4ИСИП-519

Преподаватели: Аксенова Т. Г.

Морозова М. В.

Дата: 10.03.2023

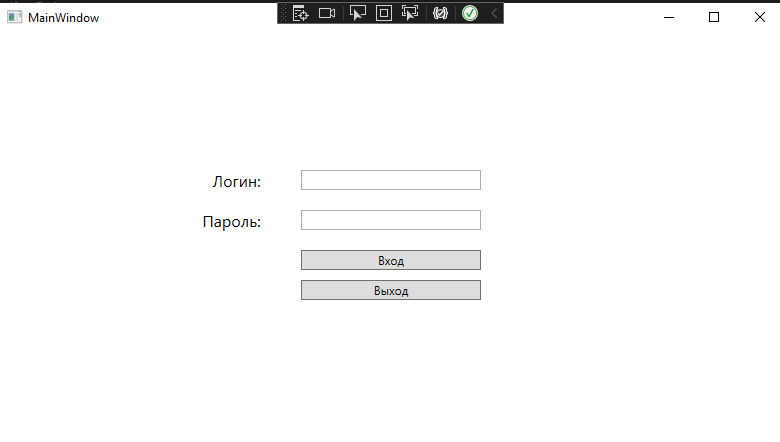


Рисунок 1. Autoregistration

Код Xaml:

<Page x:Class="Zhiganov\_519.Pages.Autoregistration"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Autoregistration">

<Grid Background="Wgile">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="\*"/>

<ColumnDefinition Width="\*"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="1\*"/>

<RowDefinition Height="1\*"/>

<RowDefinition Height="1\*"/>

</Grid.RowDefinitions>

<TextBox x:Name="Log" Height="20" Width="180" Grid.Row="1" Grid.Column="1" HorizontalAlignment="Center" VerticalAlignment="Top"/>

<TextBlock x:Name="TextLog" Text="Логин:" Grid.Row="1" Grid.Column="0" HorizontalAlignment="Right" VerticalAlignment="Top" FontSize="16" Height="21" Width="48" />

<PasswordBox x:Name="Pass" Height="20" Width="180" Grid.Row="1" Grid.Column="1" HorizontalAlignment="Center" VerticalAlignment="Top" Margin="0,40,0,0"/>

<TextBlock x:Name="TextPass" Text="Пароль:" HorizontalAlignment="Right" VerticalAlignment="Top" FontSize="16" Grid.Row="1" Grid.Column="0" Height="21" Width="58" Margin="0,40,0,0"/>

<Button x:Name="Reg" Content="Вход" Grid.Row="1" Grid.Column="1" Height="20" VerticalAlignment="Top" Margin="0,80,0,0" Width="180" Click="Reg\_Click" />

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_519.Pages

{

/// <summary>

/// Логика взаимодействия для Autoregistration.xaml

/// </summary>

public partial class Autoregistration : Page

{

public Autoregistration()

{

InitializeComponent();

}

public int a = 0;

public string s = "";

private void Back\_Click(object sender, RoutedEventArgs e)

{

}

private void Reg\_Click(object sender, RoutedEventArgs e)

{

Sotrudniki asd = DataBaseConnection.entites.Sotrudniki.Where(i => i.Login == Log.Text && i.Password == Pass.Password).FirstOrDefault();

if (a != 2)

{

if (asd != null)

{

switch (asd.Role)

{

case "Менеджер С":

NavigationService.Navigate(new Pages.Menedjer());

break;

}

}

else

{

MessageBox.Show("Проверьте введенные данные");

a++;

}

}

else

{

Grid.Visibility = Visibility;

TextPass.Visibility = Visibility.Hidden;

TextLog.Visibility = Visibility.Hidden;

Random r = new Random();

for (int i = 0; i < 5; i++)

s += (char)(r.Next(1072, 1104));

Kapcha.Text = s;

}

}

private void but1\_Click(object sender, RoutedEventArgs e)

{

if (UserText.Text == s)

{

Grid.Visibility = Visibility.Hidden;

TextPass.Visibility = Visibility.Visible;

TextLog.Visibility = Visibility.Visible;

a = 0;

s = "";

Kapcha.Text = "";

}

else

{

s = "";

Random r = new Random();

for (int i = 0; i < 5; i++)

s += (char)(r.Next(1072, 1104));

Kapcha.Text = s;

}

}

}

}

<Button x:Name="Back" Content="Выход" Grid.Row="1" Grid.Column="1" Height="20" VerticalAlignment="Top" Margin="0,110,0,0" Width="180" Click="Back\_Click"/>

<Grid x:Name="Grid" Visibility="Hidden" Grid.Row="1" Grid.Column="1" Background="White">

<TextBlock Text="Капча" HorizontalAlignment="Center" VerticalAlignment="Top" FontSize="16"/>

<TextBlock x:Name="Kapcha" Height="40" Width="200" VerticalAlignment="Top" Margin="25" FontSize="20" TextAlignment="Center"/>

<TextBox x:Name="UserText" Height="20" Width="200" VerticalAlignment="Bottom" Margin="0,0,0,50"/>

<Button x:Name="but1" Height="20" Width="200" VerticalAlignment="Bottom" Margin="10" Content="Проверить" Click="but1\_Click"/>

</Grid>

</Grid>

</Page>

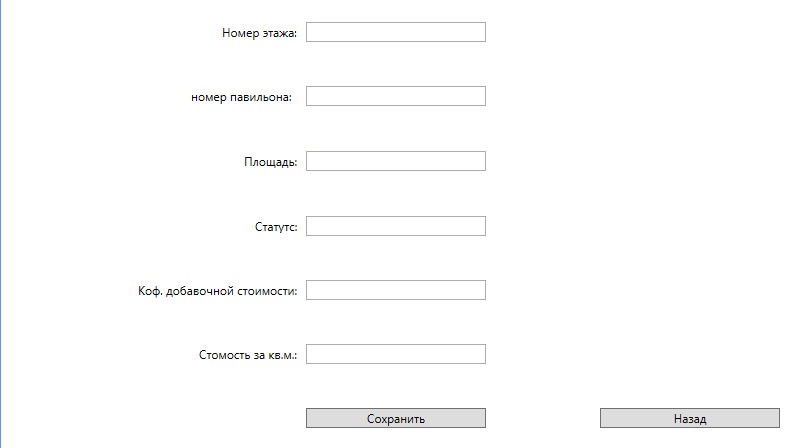


Рисунок 2. Add\_Redactoror\_Pavilion

Код xaml:

<Page x:Class=" Zhiganov\_519.Pages.Add\_Redactoror\_Pavilion"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Add\_Redactoror\_Pavilion" >

<Grid Background="White">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="29\*"/>

<ColumnDefinition Width="51\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="64\*"/>

<RowDefinition Height="28\*"/>

<RowDefinition Height="37\*"/>

<RowDefinition Height="64\*"/>

<RowDefinition Height="64\*"/>

<RowDefinition Height="64\*"/>

<RowDefinition Height="64\*"/>

<RowDefinition Height="64\*"/>

</Grid.RowDefinitions>

<TextBlock Text="Номер этажа:" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="0" Margin="221,0,0,0" Grid.ColumnSpan="2" Height="16" Width="75"/>

<TextBox Text="{Binding etaj}" x:Name="number\_etaj" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="1" Margin="15,0,0,0"/>

<TextBlock Text="номер павильона:" Grid.Row="1" VerticalAlignment="Top" HorizontalAlignment="Right" Grid.Column="0" Margin="0,24,0,0" Grid.RowSpan="2" Height="16" Width="100"/>

<TextBox Text="{Binding nomer\_paviliona}" x:Name="Number\_Pavilion" Height="20" Width="180" VerticalAlignment="Top" HorizontalAlignment="Left" Grid.Row="1" Grid.Column="1" Margin="15,22,0,0" Grid.RowSpan="2"/>

<TextBlock Text="Площадь:" Grid.Row="3" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="0" Margin="243,0,0,0" Grid.ColumnSpan="2" Height="16" Width="53"/>

<TextBox Text="{Binding plochad}" x:Name="plochad" Grid.Row="3" Grid.Column="1" Width="180" VerticalContentAlignment="Center" HorizontalAlignment="Left" Margin="15,22,0,22"/>

<TextBlock Text="Статутс:" Grid.Row="4" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="0" Margin="254,0,0,0" Grid.ColumnSpan="2" Height="15" Width="42"/>

<TextBox Text="{Binding status}" x:Name="Status" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="4" Grid.Column="1" Margin="15,0,0,0"/>

<TextBlock Text="Коф. добавочной стоимости:" Grid.Row="5" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="0" Margin="137,0,0,0" Grid.ColumnSpan="2" Height="16" Width="159"/>

<TextBox Text="{Binding kof\_dobav\_stoimosti}" x:Name="kof\_stoimosti" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="5" Grid.Column="1" Margin="15,0,0,0"/>

<TextBlock Text="Стомость за кв.м.:" Grid.Row="6" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Column="0" Margin="198,0,0,0" Grid.ColumnSpan="2" Height="16" Width="98"/>

<TextBox Text="{Binding stoimost\_za\_kv}" x:Name="Stoimost\_za\_kv" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="6" Grid.Column="1" Margin="15,0,0,0"/>

<Button Content="Сохранить" x:Name="Save" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="7" Grid.Column="1" Margin="15,0,0,0" Click="Save\_Click"/>

<Button Content="Назад" x:Name="Back" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="7" Grid.Column="1" Margin="309,0,0,0" Click="Back\_Click"/>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_519.Pages

{

/// <summary>

/// Логика взаимодействия для Add\_Redactoror\_Pavilion.xaml

/// </summary>

public partial class Add\_Redactoror\_Pavilion : Page

{

private Pavilion \_currentPavilion = new Pavilion();

public Add\_Redactoror\_Pavilion(Pavilion selectedPavilion)

{

InitializeComponent();

if (selectedPavilion != null)

\_currentPavilion = selectedPavilion;

DataContext = \_currentPavilion;

}

private void Save\_Click(object sender, RoutedEventArgs e)

{

if (\_currentPavilion.ID == 0)

DataBaseConnection.entites.Pavilion.Add(\_currentPavilion);

try

{

DataBaseConnection.entites.SaveChanges();

MessageBox.Show("ок");

NavigationService.Navigate(new Pages.Menedjer());

}

catch

{

MessageBox.Show("no ok");

}

}

private void Back\_Click(object sender, RoutedEventArgs e)

{

NavigationService.GoBack();

}

}

}

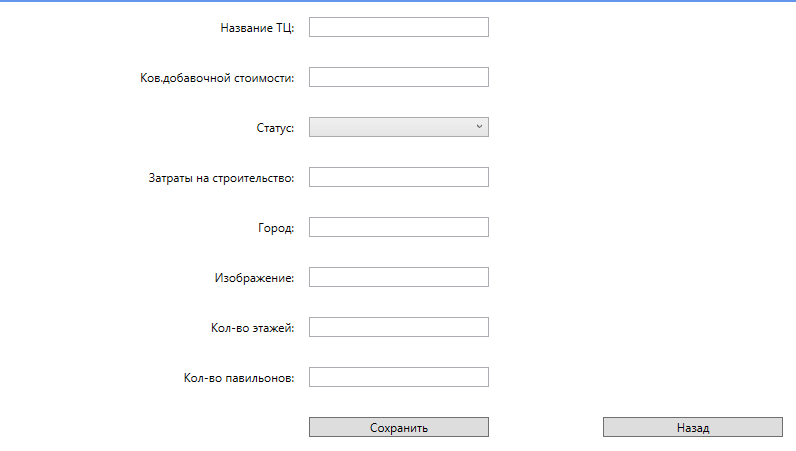


Рисунок 3. Add\_Redactoror\_TC

Код Xaml:

<Page x:Class=" Zhiganov\_519.Pages.Add\_Redactoror\_TC"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace: Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Add\_Redactoror\_TC">

<Grid Background="White">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="37\*"/>

<ColumnDefinition Width="63\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<TextBlock Text="Название ТЦ:" Grid.Row="0" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Nazvanie}" x:Name="Name\_Tc" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="0" Grid.Column="1" Margin="15"/>

<TextBlock Text="Ков.добавочной стоимости:" Grid.Row="1" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Kof\_Dobav\_Stoimosti}" x:Name="Kof\_Stoim" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="1" Grid.Column="1" Margin="15"/>

<TextBlock Text="Статус:" Grid.Row="2" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<ComboBox Text="{Binding Status}" x:Name="status" Grid.Row="2" Grid.Column="1" Height="20" Width="180" VerticalContentAlignment="Center" HorizontalAlignment="Left" Margin="15">

<ComboBoxItem>

<TextBlock Text="План" Height="20" Width="auto"/>

</ComboBoxItem>

<ComboBoxItem>

<TextBlock Text="Строительство" Height="20" Width="auto"/>

</ComboBoxItem>

<ComboBoxItem>

<TextBlock Text="Реализация" Height="20" Width="auto"/>

</ComboBoxItem>

</ComboBox>

<TextBlock Text="Затраты на строительство:" Grid.Row="3" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Stoimost}" x:Name="Zatrat" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="3" Grid.Column="1" Margin="15"/>

<TextBlock Text="Город:" Grid.Row="4" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Gorod}" x:Name="City" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="4" Grid.Column="1" Margin="15"/>

<TextBlock Text="Изображение:" Grid.Row="5" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Fhoto}" x:Name="Image" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="5" Grid.Column="1" Margin="15"/>

<TextBlock Text="Кол-во этажей:" Grid.Row="6" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Etajnost}" x:Name="Etaj" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="6" Grid.Column="1" Margin="15"/>

<TextBlock Text="Кол-во павильонов:" Grid.Row="7" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Column="0"/>

<TextBox Text="{Binding Kol\_Pavilionov}" x:Name="kol\_pavilionov" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="7" Grid.Column="1" Margin="15"/>

<Button Content="Сохранить" x:Name="Save" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Left" Grid.Row="8" Grid.Column="1" Margin="15" Click="Save\_Click"/>

<Button Content="Назад" x:Name="Back" Height="20" Width="180" VerticalAlignment="Center" HorizontalAlignment="Right" Grid.Row="8" Grid.Column="1" Margin="15" Click="Back\_Click"/>

</Grid>

</Page>

Код Cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_519.Pages

{

/// <summary>

/// Логика взаимодействия для Add\_Redactoror\_TC.xaml

/// </summary>

public partial class Add\_Redactoror\_TC : Page

{

private Tc \_currentTC = new Tc();

public string title = "";

public Add\_Redactoror\_TC(Tc selectedTC)

{

InitializeComponent();

if (selectedTC != null)

\_currentTC = selectedTC;

title = selectedTC.Nazvanie;

DataContext = \_currentTC;

}

private void Back\_Click(object sender, RoutedEventArgs e)

{

NavigationService.GoBack();

}

private void Save\_Click(object sender, RoutedEventArgs e)

{

if (status.Text == "План")

{

var vart = DataBaseConnection.entites.Pavilion.Where(i => i.nazvanie\_Tc == title).FirstOrDefault();

if (vart != null)

{

if (vart.status == "забронирован")

{

MessageBox.Show("Есть павильоны со статусом: Забронирован ");

}

}

else

{

try

{

if (\_currentTC.Etajnost == null)

DataBaseConnection.entites.Tc.Add(\_currentTC);

DataBaseConnection.entites.SaveChanges();

MessageBox.Show("ок");

NavigationService.Navigate(new Pages.Menedjer());

}

catch

{

MessageBox.Show("no ok");

}

}

}

else

{

try

{

if (\_currentTC.Etajnost == null)

DataBaseConnection.entites.Tc.Add(\_currentTC);

DataBaseConnection.entites.SaveChanges();

MessageBox.Show("ок");

NavigationService.Navigate(new Pages.Menedjer());

}

catch

{

MessageBox.Show("no ok");

}

}

}

}

}

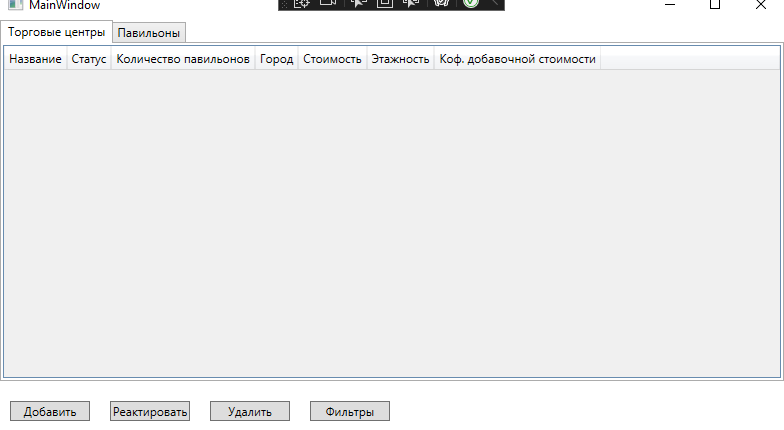


Рисунок 4. Menedjer

Код Xaml:

<Page x:Class="Zhiganov\_519.Pages.Menedjer"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Menedjer">

<Grid Background="White">

<TabControl Margin="0,0,0,50" Name="Tab">

<TabItem Header="Торговые центры">

<DataGrid x:Name="grid\_TC" AutoGenerateColumns="False" IsReadOnly="True">

<DataGrid.Columns>

<DataGridTextColumn Binding="{Binding Nazvanie}" Header="Название" Width="auto"/>

<DataGridTextColumn Binding="{Binding Status}" Header="Статус" Width="auto"/>

<DataGridTextColumn Binding="{Binding Kol\_Pavilionov}" Header="Количество павильонов" Width="auto"/>

<DataGridTextColumn Binding="{Binding Gorod}" Header="Город" Width="auto"/>

<DataGridTextColumn Binding="{Binding Stoimost}" Header="Стоимость" Width="auto"/>

<DataGridTextColumn Binding="{Binding Etajnost}" Header="Этажность" Width="auto"/>

<DataGridTextColumn Binding="{Binding Kof\_Dobav\_Stoimosti}" Header="Коф. добавочной стоимости" Width="auto"/>

</DataGrid.Columns>

</DataGrid>

</TabItem>

<TabItem Header="Павильоны">

<DataGrid x:Name="Grid\_pavilion" IsReadOnly="True" AutoGenerateColumns="False">

<DataGrid.Columns>

<DataGridTextColumn Binding="{Binding Tc.Status}" Header="Статус ТЦ" Width="auto"/>

<DataGridTextColumn Binding="{Binding nazvanie\_Tc}" Header="Название ТЦ" Width="auto"/>

<DataGridTextColumn Binding="{Binding etaj}" Header="Этаж" Width="auto"/>

<DataGridTextColumn Binding="{Binding nomer\_paviliona}" Header="Номер павильона" Width="auto"/>

<DataGridTextColumn Binding="{Binding plochad}" Header="Площадь" Width="auto"/>

<DataGridTextColumn Binding="{Binding status}" Header="Статус" Width="auto"/>

<DataGridTextColumn Binding="{Binding kof\_dobav\_stoimosti}" Header="Коф.добавочной стоимости" Width="auto"/>

<DataGridTextColumn Binding="{Binding stoimost\_za\_kv}" Header="Стоимость за кв.м." Width="auto"/>

</DataGrid.Columns>

</DataGrid>

</TabItem>

</TabControl>

<Button x:Name="Add" Content="Добавить" HorizontalAlignment="Left" VerticalAlignment="Bottom" Height="20" Width="80" Margin="10,0,0,10" Click="Add\_Click"/>

<Button x:Name="Redac" Content="Реактировать" HorizontalAlignment="Left" VerticalAlignment="Bottom" Height="20" Width="80" Margin="110,0,0,10" Click="Redac\_Click"/>

<Button x:Name="Delete" Content="Удалить" HorizontalAlignment="Left" VerticalAlignment="Bottom" Height="20" Width="80" Margin="210,0,0,10" Click="Delete\_Click"/>

<Button x:Name="Requst" Content="Фильтры" HorizontalAlignment="Left" VerticalAlignment="Bottom" Height="20" Width="80" Margin="310,0,0,10" Click="Requst\_Click"/>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespaceZhiganov\_519.Pages

{

/// <summary>

/// Логика взаимодействия для Menedjer.xaml

/// </summary>

public partial class Menedjer : Page

{

public Menedjer()

{

InitializeComponent();

grid\_TC.ItemsSource = DataBaseConnection.entites.Tc.ToList();

Grid\_pavilion.ItemsSource = DataBaseConnection.entites.Pavilion.ToList();

}

private void Delete\_Click(object sender, RoutedEventArgs e)

{

switch (Tab.SelectedIndex)

{

case 0:

var delete0 = grid\_TC.SelectedItem as Tc;

if (delete0 != null)

{

DataBaseConnection.entites.Tc.Remove(delete0);

DataBaseConnection.entites.SaveChanges();

grid\_TC.ItemsSource = DataBaseConnection.entites.Tc.ToList();

}

break;

case 1:

var delete1 = Grid\_pavilion.SelectedItem as Pavilion;

if (delete1 != null)

{

if (delete1.status == "забронирован")

{

MessageBox.Show("Невозможно удалить. статус: Забронирован ");

}

else if (delete1.status == "арендован")

{

MessageBox.Show("Невозможно удалить. статус: Арендован ");

}

else

{

DataBaseConnection.entites.Pavilion.Remove(delete1);

DataBaseConnection.entites.SaveChanges();

Grid\_pavilion.ItemsSource = DataBaseConnection.entites.Pavilion.ToList();

}

}

break;

}

}

private void Redac\_Click(object sender, RoutedEventArgs e)

{

switch (Tab.SelectedIndex)

{

case 0:

if (grid\_TC.SelectedItem != null)

{

NavigationService.Navigate(new Pages.Add\_Redactoror\_TC(grid\_TC.SelectedItem as Tc));

}

else

{

MessageBox.Show("Выберети элемент");

}

break;

case 1:

var redac = Grid\_pavilion.SelectedItem as Pavilion;

if (redac != null)

{

if (redac.status == "забронирован")

{

MessageBox.Show("Невозможно удалить. статус: Забронирован ");

}

else if (redac.status == "арендован")

{

MessageBox.Show("Невозможно удалить. статус: Арендован ");

}

else

{

NavigationService.Navigate(new Pages.Add\_Redactoror\_Pavilion(Grid\_pavilion.SelectedItem as Pavilion));

}

}

else

{

MessageBox.Show("Выберети элемент");

}

break;

}

}

private void Add\_Click(object sender, RoutedEventArgs e)

{

switch (Tab.SelectedIndex)

{

case 0:

NavigationService.Navigate(new Pages.Add\_Redactoror\_TC(null));

break;

case 1:

NavigationService.Navigate(new Pages.Add\_Redactoror\_Pavilion(null));

break;

}

}

private void Requst\_Click(object sender, RoutedEventArgs e)

{

switch (Tab.SelectedIndex)

{

case 0:

NavigationService.Navigate(new Pages.RequastTC());

break;

case 1:

NavigationService.Navigate(new Pages.RequastPavilion());

break;

}

}

}

}

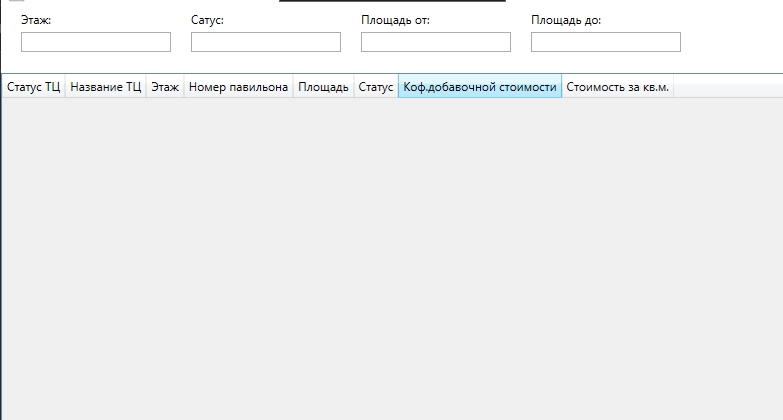


Рисунок 5. RequastPavilion

<Page x:Class="Zhiganov\_519.Pages.RequastPavilion"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="RequastPavilion">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="34\*"/>

<RowDefinition Height="191\*"/>

</Grid.RowDefinitions>

<DataGrid x:Name="Grid\_pavilion" IsReadOnly="True" AutoGenerateColumns="False" Grid.Row="1">

<DataGrid.Columns>

<DataGridTextColumn Binding="{Binding Tc.Status}" Header="Статус ТЦ" Width="auto"/>

<DataGridTextColumn Binding="{Binding nazvanie\_Tc}" Header="Название ТЦ" Width="auto"/>

<DataGridTextColumn Binding="{Binding etaj}" Header="Этаж" Width="auto"/>

<DataGridTextColumn Binding="{Binding nomer\_paviliona}" Header="Номер павильона" Width="auto"/>

<DataGridTextColumn Binding="{Binding plochad}" Header="Площадь" Width="auto"/>

<DataGridTextColumn Binding="{Binding status}" Header="Статус" Width="auto"/>

<DataGridTextColumn Binding="{Binding kof\_dobav\_stoimosti}" Header="Коф.добавочной стоимости" Width="auto"/>

<DataGridTextColumn Binding="{Binding stoimost\_za\_kv}" Header="Стоимость за кв.м." Width="auto"/>

</DataGrid.Columns>

</DataGrid>

<TextBlock Text="Этаж:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="20,0,0,0"/>

<TextBox SelectionChanged="Etaj\_SelectionChanged" x:Name="Etaj" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="20 0 0 0"/>

<TextBlock Text="Сатус:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="190,0,0,0"/>

<TextBox SelectionChanged="Status\_SelectionChanged" x:Name="Status" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="190 0 0 0"/>

<TextBlock Text="Площадь от:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="360,0,0,0"/>

<TextBox SelectionChanged="Ploshad\_ot\_SelectionChanged" x:Name="Ploshad\_ot" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="360 0 0 0"/>

<TextBlock Text="Площадь до:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="530,0,0,0"/>

<TextBox SelectionChanged="Ploshad\_do\_SelectionChanged" x:Name="Ploshad\_do" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="530 0 0 0"/>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_519.Pages

{

/// <summary>

/// Логика взаимодействия для RequastPavilion.xaml

/// </summary>

public partial class RequastPavilion : Page

{

public RequastPavilion()

{

InitializeComponent();

Grid\_pavilion.ItemsSource = DataBaseConnection.entites.Pavilion.ToList();

}

public void dc()

{

if (Status.Text.Length != 0)

{

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.status == Status.Text

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Etaj.Text.Length != 0)

{

var F = Convert.ToInt32(Etaj.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.etaj == F

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Ploshad\_ot.Text.Length != 0 && Ploshad\_do.Text.Length != 0)

{

var G = Convert.ToDouble(Ploshad\_ot.Text);

var H = Convert.ToDouble(Ploshad\_do.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.plochad >= G

where a.plochad <= H

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Etaj.Text.Length != 0 && Status.Text.Length != 0)

{

var F = Convert.ToInt32(Etaj.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.status == Status.Text

where a.etaj == F

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Status.Text.Length != 0 && Ploshad\_ot.Text.Length != 0 && Ploshad\_do.Text.Length != 0)

{

var G = Convert.ToDouble(Ploshad\_ot.Text);

var H = Convert.ToDouble(Ploshad\_do.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.status == Status.Text

where a.plochad >= G

where a.plochad <= H

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Etaj.Text.Length != 0 && Ploshad\_ot.Text.Length != 0 && Ploshad\_do.Text.Length != 0)

{

var G = Convert.ToDouble(Ploshad\_ot.Text);

var H = Convert.ToDouble(Ploshad\_do.Text);

var F = Convert.ToInt32(Etaj.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.etaj == F

where a.plochad >= G

where a.plochad <= H

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

if (Status.Text.Length != 0 && Etaj.Text.Length != 0 && Ploshad\_ot.Text.Length != 0 && Ploshad\_do.Text.Length != 0)

{

var G = Convert.ToDouble(Ploshad\_ot.Text);

var H = Convert.ToDouble(Ploshad\_do.Text);

var F = Convert.ToInt32(Etaj.Text);

var abc = (from a in DataBaseConnection.entites.Pavilion

where a.status == Status.Text

where a.etaj == F

where a.plochad >= G

where a.plochad <= H

select a).ToList();

Grid\_pavilion.ItemsSource = abc;

}

}

private void Etaj\_SelectionChanged(object sender, RoutedEventArgs e)

{

dc();

}

private void Status\_SelectionChanged(object sender, RoutedEventArgs e)

{

dc();

}

private void Ploshad\_ot\_SelectionChanged(object sender, RoutedEventArgs e)

{

dc();

}

private void Ploshad\_do\_SelectionChanged(object sender, RoutedEventArgs e)

{

dc();

}

}

}

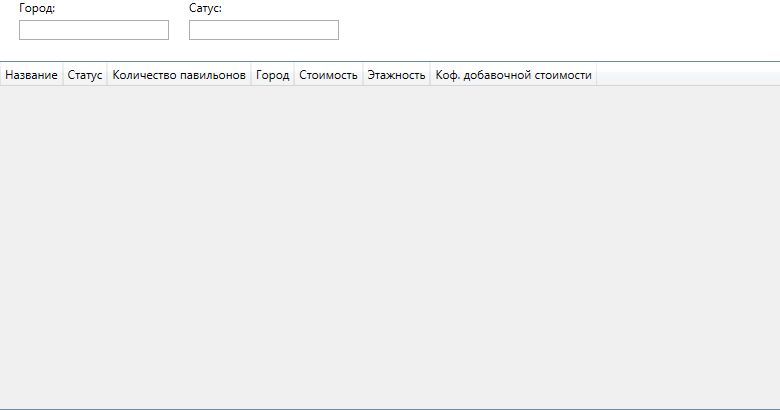


Рисунок 6. RequastTC

Код XAML:

<Page x:Class="Zhiganov\_519.Pages.RequastTC"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_519.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="RequastTC">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="34\*"/>

<RowDefinition Height="191\*"/>

</Grid.RowDefinitions>

<DataGrid x:Name="Tc" AutoGenerateColumns="False" IsReadOnly="True" Grid.Row="1">

<DataGrid.Columns>

<DataGridTextColumn Binding="{Binding Nazvanie}" Header="Название" Width="auto"/>

<DataGridTextColumn Binding="{Binding Status}" Header="Статус" Width="auto"/>

<DataGridTextColumn Binding="{Binding Kol\_Pavilionov}" Header="Количество павильонов" Width="auto"/>

<DataGridTextColumn Binding="{Binding Gorod}" Header="Город" Width="auto"/>

<DataGridTextColumn Binding="{Binding Stoimost}" Header="Стоимость" Width="auto"/>

<DataGridTextColumn Binding="{Binding Etajnost}" Header="Этажность" Width="auto"/>

<DataGridTextColumn Binding="{Binding Kof\_Dobav\_Stoimosti}" Header="Коф. добавочной стоимости" Width="auto"/>

</DataGrid.Columns>

</DataGrid>

<TextBlock Text="Город:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="20,0,0,0"/>

<TextBox SelectionChanged="Gorod\_SelectionChanged" x:Name="Gorod" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="20 0 0 0"/>

<TextBlock Text="Сатус:" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="190,0,0,0"/>

<TextBox SelectionChanged="Status\_SelectionChanged" x:Name="Status" Height="20" Width="150" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="190 0 0 0"/>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_519.Pages

{

/// <summary>

/// </summary>

public partial class RequastTC : Page

{

public RequastTC()

{

InitializeComponent();

Tc.ItemsSource = DataBaseConnection.entites.Tc.ToList();

}

private async void cd()

{

await Task.Delay(100);

if (Gorod.Text.Length != 0)

{

var abc = (from a in DataBaseConnection.entites.Tc

where a.Gorod == Gorod.Text

where a.Gorod != "удален"

select a).ToList();

Tc.ItemsSource = abc;

}

if (Status.Text.Length != 0)

{

var abc = (from a in DataBaseConnection.entites.Tc

where a.Status == Status.Text

select a).ToList();

Tc.ItemsSource = abc;

}

if (Gorod.Text.Length != 0 && Status.Text.Length != 0)

{

var abc = (from a in DataBaseConnection.entites.Tc

where a.Gorod == Gorod.Text

where a.Status == Status.Text

select a).ToList();

Tc.ItemsSource = abc;

}

}

private void Gorod\_SelectionChanged(object sender, RoutedEventArgs e)

{

cd();

}

private void Status\_SelectionChanged(object sender, RoutedEventArgs e)

{

cd();

}

}

}

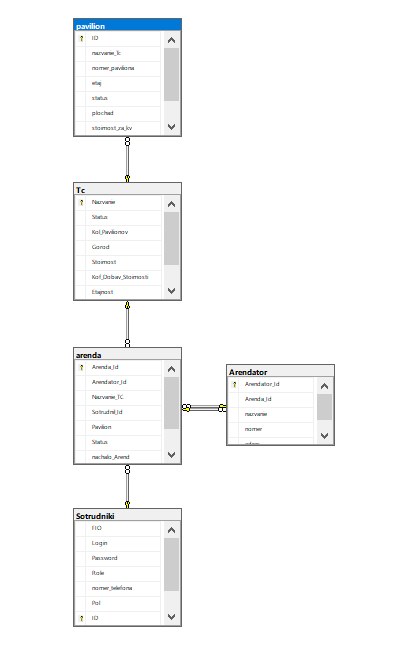


Рисунок 7. Диаграма SQL

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ №2**

# **Тема: «Товары автосервиса»**

Студенты: Жиганов В. О.

Группа: 4ИСИП-519

Преподаватели: Аксенова Т. Г.

Морозова М. В.

Дата: 10.03.2023



Рисунок 1. Add.Redactoror

Код XAML:

<Page x:Class="Zhiganov\_Vladislav\_519.Add\_Redactor"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_Vladislav \_519"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Add\_Redactor">

<Grid Background="#FFA29595">

<Grid.RowDefinitions>

<RowDefinition />

<RowDefinition />

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="1\*"/>

<ColumnDefinition Width="1\*"/>

</Grid.ColumnDefinitions>

<TextBox Text="{Binding Title}" HorizontalAlignment="Left" Grid.Row="0" Grid.Column="1" x:Name="Txt\_0" Height="20" Width="180" Margin="10"/>

<TextBox Text="{Binding Cost}" HorizontalAlignment="Left" Grid.Row="1" Grid.Column="1" x:Name="Txt\_1" Height="20" Width="180" Margin="10"/>

<TextBox Text="{Binding Description}" HorizontalAlignment="Left" Grid.Row="2" Grid.Column="1" x:Name="Txt\_2" Height="20" Width="180" Margin="10"/>

<TextBox Text="{Binding MainImagePath}" HorizontalAlignment="Left" Grid.Row="3" Grid.Column="1" x:Name="Txt\_3" Height="20" Width="180" Margin="10"/>

<TextBox Text="{Binding IsActive}" HorizontalAlignment="Left" Grid.Row="4" Grid.Column="1" x:Name="Txt\_4" Height="20" Width="180" Margin="10"/>

<TextBox Text="{Binding ManufacturerID}" HorizontalAlignment="Left" Grid.Row="5" Grid.Column="1" x:Name="Txt\_5" Height="20" Width="180" Margin="10"/>

<TextBlock Text="Название " Grid.Row="0" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<TextBlock Text="Стоимость " Grid.Row="1" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<TextBlock Text="Комментарии" Grid.Row="2" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<TextBlock Text="Изображение " Grid.Row="3" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<TextBlock Text="Есть на складе" Grid.Row="4" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<TextBlock Text="Количество" Grid.Row="5" Grid.Column="0" HorizontalAlignment="Right" Height="20" Margin="10"/>

<Button x:Name="Save" Content="Сохранить" HorizontalAlignment="Right" Height="20" Width="80" Margin="10,0,0,0" Grid.Column="0" Grid.Row="8" Click="Save\_Click"/>

<Button x:Name="Back" Content="Назад" HorizontalAlignment="Left" Height="20" Width="80" Margin="0,0,10,0" Grid.Column="1" Grid.Row="8" Click="Back\_Click"/>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_Vladislav\_519

{

/// <summary>

/// </summary>

public partial class Add\_Redactor : Page

{

Product \_product = new Product();

public Add\_Redactor(Product SelectedProduct)

{

InitializeComponent();

if(SelectedProduct != null)

{

\_product = SelectedProduct;

}

DataContext= \_product;

}

private void Save\_Click(object sender, RoutedEventArgs e)

{

try

{

if (\_product.ID == 0)

DataBaseConnection.entities.Product.Add(\_product);

DataBaseConnection.entities.SaveChanges();

MessageBox.Show("Ваше сохранение было успешно сохранено ");

}

catch { }

}

private void Back\_Click(object sender, RoutedEventArgs e)

{

NavigationService.GoBack();

}

}

}

Риссунок 2. ADD\_Manuf

Код XAML:

<Page x:Class="Zhiganov\_Vladislav\_519.ADD\_Manuf"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:Zhiganov\_Vladislav\_519"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="ADD\_Manuf">

<Grid Background="#FFA29595">

<Grid.RowDefinitions>

<RowDefinition Height="50"/>

<RowDefinition Height="10\*"/>

<RowDefinition Height="1\*"/>

</Grid.RowDefinitions>

<TextBlock Text="Название:" Grid.Row="0" Height="20" Width="auto" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="20 0 0 0"/>

<TextBox x:Name="txt\_Name\_P" Height="20" Width="150" HorizontalAlignment="Left" Margin="20 0 0 0 " SelectionChanged="txt\_Name\_P\_SelectionChanged"/>

<TextBlock Text="Описание:" Grid.Row="0" Height="20" Width="auto" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="190 0 0 0"/>

<TextBox x:Name="txt\_Description\_P" Height="20" Width="150" HorizontalAlignment="Left" Margin="190 0 0 0 " SelectionChanged="txt\_Description\_P\_SelectionChanged"/>

<TextBlock Text="Компания:" Grid.Row="0" Height="20" Width="auto" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="360 0 0 0"/>

<ComboBox x:Name="dsa" DisplayMemberPath="Name" Height="20" Width="150" Grid.Row="0" HorizontalAlignment="Left" Margin="360 0 0 0" SelectionChanged="dsa\_SelectionChanged">

<TextBlock>По умолчанию</TextBlock>

<TextBlock>Motor Oil</TextBlock>

<TextBlock>Helix</TextBlock>

<TextBlock>Super</TextBlock>

<TextBlock>Turbo SYN</TextBlock>

<TextBlock>Dexos</TextBlock>

<TextBlock>Molygen New Generation</TextBlock>

<TextBlock>Top Tec</TextBlock>

<TextBlock>ENGINE OIL</TextBlock>

<TextBlock>Mobil</TextBlock>

<TextBlock>Dexos</TextBlock>

<TextBlock>Genesis Armortech</TextBlock>

<TextBlock>Optimal Synth</TextBlock>

<TextBlock>Evolution</TextBlock>

<TextBlock>Special C G</TextBlock>

<TextBlock>Longlife III G</TextBlock>

<TextBlock>Special Tec LL</TextBlock>

<TextBlock>Dexos 2</TextBlock>

<TextBlock>Gasoline Fully</TextBlock>

<TextBlock>Formula F</TextBlock>

<TextBlock>Zepro Euro Spec SN/CF</TextBlock>

<TextBlock>SN</TextBlock>

<TextBlock>SN/GF</TextBlock>

<TextBlock>Трос буксировочный</TextBlock>

<TextBlock>Molygen</TextBlock>

<TextBlock>QUARTZ</TextBlock>

<TextBlock>Original oil</TextBlock>

<TextBlock>масло 8100</TextBlock>

<TextBlock>Magnatec</TextBlock>

<TextBlock>Щетка для снега</TextBlock>

<TextBlock>Premium DPF</TextBlock>

<TextBlock>Special G</TextBlock>

<TextBlock>Ecstar F9000</TextBlock>

<TextBlock>Genesis Armortech</TextBlock>

<TextBlock>Mobil 1</TextBlock>

<TextBlock>Genuine</TextBlock>

<TextBlock>ULTRA</TextBlock>

<TextBlock>Zepro Euro Spec</TextBlock>

<TextBlock>76</TextBlock>

<TextBlock>Top Tec</TextBlock>

<TextBlock>Mobil 1</TextBlock>

<TextBlock>Leichtlauf Special</TextBlock>

<TextBlock>Helix HX8</TextBlock>

<TextBlock>Fuel Economy</TextBlock>

<TextBlock>X9</TextBlock>

<TextBlock>EDGE Professional</TextBlock>

<TextBlock>Magnatec</TextBlock>

<TextBlock>Supreme Synthetic</TextBlock>

<TextBlock>Люкс</TextBlock>

<TextBlock>Жилет светоотражающий</TextBlock>

<TextBlock>Premium LF</TextBlock>

<TextBlock>Zepro Touring</TextBlock>

<TextBlock>Кредитница</TextBlock>

</ComboBox>

<ComboBox x:Name="txt\_App\_Bot" Height="20" Width="150" HorizontalAlignment="Left" Margin="530 0 0 0" SelectionChanged="txt\_App\_Bot\_SelectionChanged">

<ComboBoxItem>

<TextBlock Text="По возрастанию"/>

</ComboBoxItem>

<ComboBoxItem>

<TextBlock Text="По убыванию"/>

</ComboBoxItem>

</ComboBox>

<Button x:Name="Delete" Grid.Row="2" Height="20" Width="100" Content="Удалить" HorizontalAlignment="Center" Margin="-100 0 0 0" Click="Delete\_Click"/>

<Button x:Name="Redactor" Grid.Row="2" Height="20" Width="100" Content="Редактировать" HorizontalAlignment="Center" Margin="-500 0 0 0" Click="Redactor\_Click"/>

<Button x:Name="Add" Grid.Row="2" Height="20" Width="100" Content="Добавить" HorizontalAlignment="Center" Margin="360 0 0 0" Click="Add\_Click"/>

<ListView Name="ListUser" Grid.Row="1" ScrollViewer.HorizontalScrollBarVisibility="Disabled" HorizontalContentAlignment="Center">

<ListView.ItemContainerStyle>

<Style TargetType="ListViewItem">

<Setter Property="Background" Value="White"></Setter>

<Style.Triggers>

<DataTrigger Binding="{Binding IsActive}" Value="False">

<Setter Property="Background" Value="Gray"/>

</DataTrigger>

</Style.Triggers>

</Style>

</ListView.ItemContainerStyle>

<ListView.ItemsPanel>

<ItemsPanelTemplate>

<WrapPanel Orientation="Horizontal" HorizontalAlignment="Center"></WrapPanel>

</ItemsPanelTemplate>

</ListView.ItemsPanel>

<ListView.ItemTemplate>

<DataTemplate>

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="1\*"></RowDefinition>

<RowDefinition Height="10\*"></RowDefinition>

<RowDefinition Height="1\*"></RowDefinition>

</Grid.RowDefinitions>

<Image Width="250" Grid.Row="1" Stretch="UniformToFill" HorizontalAlignment="Center">

<Image.Source>

<Binding Path="MainImagePath">

<Binding.TargetNullValue>

<ImageSource>/Товары автосервиса/0070C563.jpg</ImageSource>

</Binding.TargetNullValue>

</Binding>

</Image.Source>

</Image>

<TextBlock TextAlignment="Center" Width="auto" Grid.Row="0" TextWrapping="Wrap" Text="{Binding Title}" FontSize="18" FontWeight="Bold"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="auto" Grid.Row="2" HorizontalAlignment="Left" FontSize="18" Text="{Binding Cost}"/>

<TextBlock Width="auto" Grid.Row="2" HorizontalAlignment="Right" FontSize="18" Text="{Binding IsActive}"/>

</Grid>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</Grid>

</Page>

Код CS:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Zhiganov\_Vladislav\_519

{

public partial class ADD\_Manuf : Page

{

public ADD\_Manuf()

{

InitializeComponent();

dsa.ItemsSource = DataBaseConnection.entities.Manufacturer.ToList();

txt\_App\_Bot.SelectedIndex = 0;

var requst0 = (from a in DataBaseConnection.entities.Product

orderby a.Cost

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

dsa.SelectedIndex = 0;

var currentUsers = derevnyEntities.GetContext().Product.ToList();

ListUser.ItemsSource = currentUsers;

UpdateUsers();

}

private void txt\_Name\_P\_SelectionChanged(object sender, RoutedEventArgs e)

{

UpdateUsers();

}

private void txt\_Description\_P\_SelectionChanged(object sender, RoutedEventArgs e)

{

ca();

}

private void dsa\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

ca();

}

private void txt\_App\_Bot\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

ca();

}

private async void ca()

{

if (txt\_Name\_P.Text.Length == 0 && txt\_Description\_P.Text.Length == 0 && dsa.Text.Length == 0 && txt\_App\_Bot.Text.Length == 0)

{

ListUser.ItemsSource = DataBaseConnection.entities.Product.ToList();

}

await Task.Delay(100);

if (txt\_Name\_P.Text.Length != 0)

{

var requst0 = (from a in DataBaseConnection.entities.Product

where a.Title == txt\_Name\_P.Text

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

else if (txt\_Description\_P.Text.Length != 0)

{

var requst0 = (from a in DataBaseConnection.entities.Product

where a.Description == txt\_Description\_P.Text

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

else if (dsa.Text.Length != 0)

{

var requst0 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

var requst1 = (from b in DataBaseConnection.entities.Product

where b.ManufacturerID == requst0

select b).ToList();

ListUser.ItemsSource = requst1.ToList();

}

else if (txt\_App\_Bot.Text.Length != 0)

{

if (txt\_App\_Bot.Text == "По возрастанию")

{

var requst0 = (from a in DataBaseConnection.entities.Product

orderby a.Cost

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

if (txt\_App\_Bot.Text == "По убыванию")

{

var requst0 = DataBaseConnection.entities.Product.OrderByDescending(a => a.Cost).ToList();

ListUser.ItemsSource = requst0.ToList();

}

}

if (txt\_Name\_P.Text.Length != 0 && txt\_Description\_P.Text.Length != 0)

{

var requst0 = (from a in DataBaseConnection.entities.Product

where a.Title == txt\_Name\_P.Text

where a.Description == txt\_Description\_P.Text

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

if (txt\_Name\_P.Text.Length != 0 && txt\_Description\_P.Text.Length != 0 && dsa.Text.Length != 0)

{

var requst0 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

var requst1 = (from a in DataBaseConnection.entities.Product

where a.Title == txt\_Name\_P.Text

where a.Description == txt\_Description\_P.Text

where a.ManufacturerID == requst0

select a).ToList();

ListUser.ItemsSource = requst1.ToList();

}

if (txt\_Name\_P.Text.Length != 0 && txt\_Description\_P.Text.Length != 0 && dsa.Text.Length != 0 && txt\_App\_Bot.Text.Length != 0)

{

var requst01 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

if (txt\_App\_Bot.Text == "По возрастанию")

{

var requst0 = (from a in DataBaseConnection.entities.Product

orderby a.Cost

where a.Title == txt\_Name\_P.Text

where a.Description == txt\_Description\_P.Text

where a.ManufacturerID == requst01

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

if (txt\_App\_Bot.Text == "По убыванию")

{

var requst0 = DataBaseConnection.entities.Product.OrderByDescending(a => a.Cost).ToList();

ListUser.ItemsSource = requst0.Where(a => a.Title == txt\_Name\_P.Text && a.Description == txt\_Description\_P.Text && a.ManufacturerID == requst01).ToList();

}

}

if (txt\_Description\_P.Text.Length != 0 && dsa.Text.Length != 0)

{

var requst01 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

var requst1 = (from b in DataBaseConnection.entities.Product

where b.ManufacturerID == requst01

where b.Description == txt\_Description\_P.Text

select b).ToList();

ListUser.ItemsSource = requst1;

}

if (txt\_Description\_P.Text.Length != 0 && dsa.Text.Length != 0 && txt\_App\_Bot.Text.Length != 0)

{

var requst01 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

if (txt\_App\_Bot.Text == "По возрастанию")

{

var requst0 = (from a in DataBaseConnection.entities.Product

orderby a.Cost

where a.ManufacturerID == requst01

where a.Description == txt\_Description\_P.Text

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

if (txt\_App\_Bot.Text == "По убыванию")

{

var requst0 = DataBaseConnection.entities.Product.OrderByDescending(a => a.Cost).ToList();

ListUser.ItemsSource = requst0.Where(a => a.ManufacturerID == requst01 && a.Description == txt\_Description\_P.Text).ToList();

}

}

if (dsa.Text.Length != 0 && txt\_App\_Bot.Text.Length != 0)

{

var requst01 = (from a in DataBaseConnection.entities.Manufacturer

where a.Name == dsa.Text

select a.ID).FirstOrDefault();

if (txt\_App\_Bot.Text == "По возрастанию")

{

var requst0 = (from a in DataBaseConnection.entities.Product

orderby a.Cost

where a.ManufacturerID == requst01

select a).ToList();

ListUser.ItemsSource = requst0.ToList();

}

if (txt\_App\_Bot.Text == "По убыванию")

{

var requst0 = DataBaseConnection.entities.Product.OrderByDescending(a => a.Cost).ToList();

ListUser.ItemsSource = requst0.Where(a => a.ManufacturerID == requst01).ToList();

}

}

}

private void Delete\_Click(object sender, RoutedEventArgs e)

{

var delete0 = ListUser.SelectedItem as Product;

if (delete0 != null)

{

DataBaseConnection.entities.Product.Remove(delete0);

DataBaseConnection.entities.SaveChanges();

ca();

MessageBox.Show("Успешно");

}

}

private void Redactor\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Add\_Redactor(ListUser.SelectedItem as Product));

}

private void Add\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Add\_Redactor(null));

}

private void UpdateUsers()

{

var currentUsers = derevnyEntities.GetContext().Product.ToList();

currentUsers = currentUsers.Where(x => x.Title.ToLower().Contains(txt\_Name\_P.Text.ToLower())).ToList();

if (txt\_App\_Bot.SelectedIndex != 0)

currentUsers = currentUsers.Where(x => x.ManufacturerID == txt\_App\_Bot.SelectedIndex).ToList();

if (txt\_App\_Bot.SelectedIndex == 0)

ListUser.ItemsSource = currentUsers.OrderBy(x => x.Cost).ToList();

else

ListUser.ItemsSource = currentUsers.OrderByDescending(x => x.Cost).ToList();

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ №3**

# **Тема: «ГИБДД»**

Студенты: Жиганов В. О.

Группа: 4ИСИП-519

Преподаватель: Аксенова Т. Г.

Дата: 12.03.2023

****

Рисунок 1. Autregistration

<Page x:Class=”GiBdd\_Zhiganov\_Vlad.Pages.Autregistration”

xmlns=”http://schemas.microsoft.com/winfx/2006/xaml/presentation”

xmlns:x=”http://schemas.microsoft.com/winfx/2006/xaml”

xmlns:mc=”http://schemas.openxmlformats.org/markup-compatibility/2006”

xmlns:d=”http://schemas.microsoft.com/expression/blend/2008”

xmlns:local=”clr-namespace:GiBdd\_ Zhiganov\_Vlad.Pages”

mc:Ignorable=”d”

d:DesignHeight=”450” d:DesignWidth=”800”

Title=”Autregistration”>

<Grid Background=”#FFB5D5CA”>

<Grid.RowDefinitions>

<RowDefinition Height=”193\*”/>

<RowDefinition Height=”153\*”/>

<RowDefinition Height=”104\*”/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width=”143\*”/>

<ColumnDefinition Width=”657\*”/>

<ColumnDefinition Width=”0\*”/>

</Grid.ColumnDefinitions>

<Label Content=”Логин” Margin=”0,25.2,1.2,0” VerticalAlignment=”Top” Height=”38” Grid.Column=”0” Grid.Row=”1” HorizontalContentAlignment=”Right”/>

<Label Content=”Пароль” Margin=”0,82.2,1.2,0” VerticalAlignment=”Top” Height=”37” Grid.Column=”0” Grid.Row=”1” HorizontalContentAlignment=”Right”/>

<TextBox x:Name =”TextBoxLogin” ToolTip=”Введите логин” Margin=”4.8,25.2,0,0” Background=”Transparent” VerticalAlignment=”Top” Width=”213” HorizontalAlignment=”Left” Height=”38” Grid.Column=”1” Grid.Row=”1”/>

<PasswordBox x:Name =”PasswordBox” Background=”Transparent” ToolTip=”Введите пароль” HorizontalAlignment=”Left” Margin=”4.8,82.2,0,0” VerticalAlignment=”Top” Width=”213” Height=”37” Grid.Column=”1” Grid.Row=”1”/>

<Button Content=”Вход” Click=”Button\_Click” ToolTip=”Вход в систему” HorizontalAlignment=”Left” Margin=”10.8,10.4,0,0” VerticalAlignment=”Top” Width=”160” Height=”35” Grid.Column=”1” Grid.Row=”2”/>

<Image Source=”/GiBdd\_ Zhiganov\_Vlad;component/GiBdd\_ Zhiganov\_Vlad\_logo.png” Stretch=”Fill” Margin=”11,0,0,20” Grid.Column=”1” HorizontalAlignment=”Left” Width=”246”/>

</Grid>

</Page>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace GiBdd\_Zhiganov\_Vlad.Pages

{

/// <summary>

/// </summary>

public partial class AuthPage : Page

{

public AuthPage()

{

InitializeComponent();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

if (string.IsNullOrEmpty(TextBoxLogin.Text) || string.IsNullOrEmpty(PasswordBox.Password))

{

MessageBox.Show("Введите логин или пароль");

return;

}

using (var db = new Entities1())

{

var user = db.Employess

.AsNoTracking()

.FirstOrDefault(u => u.Login == TextBoxLogin.Text && u.Password == PasswordBox.Password);

if (user == null)

{

MessageBox.Show("Пользователь с такими данными не найден");

return;

}

MessageBox.Show("Пользователь успешно найден!");

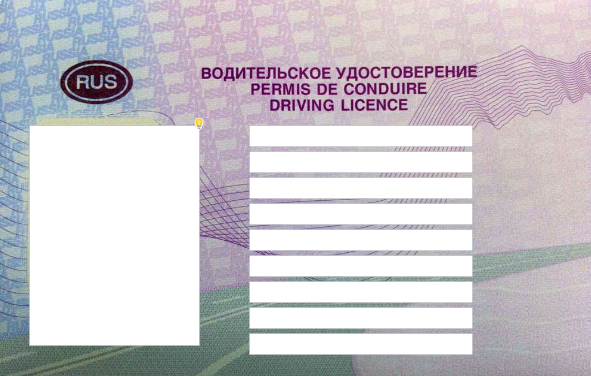
NavigationService?.Navigate(new Page1());

}

}

}

}

  
Рисунок 2. Card

<Page x:Class="GiBdd\_ Zhiganov\_Vlad.Pages.Card"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:GiBdd\_ Zhiganov\_Vlad.Pages"

mc:Ignorable="d"

Title="Card" Height="565.84" Width="894.678">

<Grid>

<Image Source="/GiBdd\_ Zhiganov\_Vlad;component/driver\_license\_template.jpg" Stretch="Fill"/>

<TextBox Text="{Binding name}" HorizontalAlignment="Left" Height="34" Margin="376,187,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding middlename}" HorizontalAlignment="Left" Height="34" Margin="376,226,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding passportserial}" HorizontalAlignment="Left" Height="34" Margin="376,265,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding passportnumber}" HorizontalAlignment="Left" Height="34" Margin="376,304,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding address}" HorizontalAlignment="Left" Height="34" Margin="376,343,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding phone}" HorizontalAlignment="Left" Height="34" Margin="376,382,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding licence\_date}" HorizontalAlignment="Left" Height="34" Margin="376,421,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding expire\_date}" HorizontalAlignment="Left" Height="34" Margin="376,460,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<TextBox Text="{Binding categories}" HorizontalAlignment="Left" Height="34" Margin="376,499,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="336"/>

<ListView x:Name="ListUser" ScrollViewer.HorizontalScrollBarVisibility="Disabled" HorizontalContentAlignment="Center" HorizontalAlignment="Left" Height="332" Margin="47,187,0,0" VerticalAlignment="Top" Width="256">

<ListView.ItemsPanel>

<ItemsPanelTemplate>

<WrapPanel Orientation="Horizontal" HorizontalAlignment="Center"/>

</ItemsPanelTemplate>

</ListView.ItemsPanel>

<ListView.ItemTemplate>

<DataTemplate>

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="auto"/>

<RowDefinition Height="450"/>

<RowDefinition Height="auto"/>

</Grid.RowDefinitions>

<Image Width="450" Grid.Row="1" Stretch="UniformToFill" HorizontalAlignment="Center">

<Image.Source>

<Binding Path="Photo">

<Binding.TargetNullValue>

<ImageSource>/photo/001-happy-18.png</ImageSource>

</Binding.TargetNullValue>

</Binding>

</Image.Source>

</Image>

</Grid>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</Grid>

</Page>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace GiBdd\_Zhiganov\_Vlad.Pages

{

/// <summary>

/// </summary>

public partial class Card : Page

{

private Drivers \_currentUser = new Drivers();

public Card(Drivers selectedUser)

{

InitializeComponent();

if (selectedUser != null)

\_currentUser = selectedUser;

DataContext = \_currentUser;

}

}

}

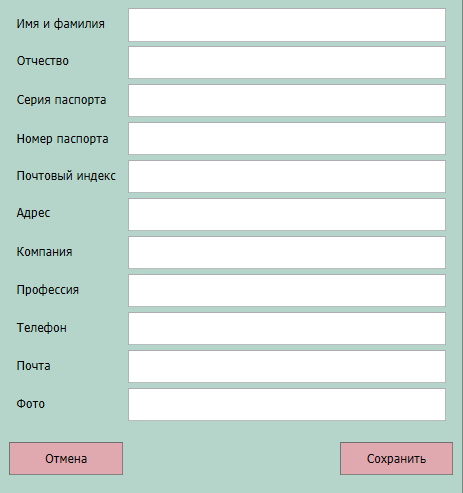


Рисунок 3. New Drivers

<Page x:Class="GiBdd\_ Zhiganov\_Vlad.Pages.AddUserPage"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:GiBdd\_ Zhiganov\_Vlad.Pages"

mc:Ignorable="d"

Title="AddUserPage" Height="537.188" Width="500">

<Grid Background="#FFB5D5CA">

<Grid.RowDefinitions>

<RowDefinition/>

</Grid.RowDefinitions>

<Button x:Name="ButtonCancel" Click="ButtonCancel\_Click" Content="Отмена" HorizontalAlignment="Left" Margin="10,479,0,0" VerticalAlignment="Top" Width="123"/>

<Button x:Name="ButtonSave" Click="ButtonSave\_Click" Content="Сохранить" HorizontalAlignment="Left" Margin="367,479,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Имя и фамилия" HorizontalAlignment="Left" Margin="10,10,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Oтчество" HorizontalAlignment="Left" Margin="10,50,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Серия паспорта" HorizontalAlignment="Left" Margin="10,92,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Номер паспорта" HorizontalAlignment="Left" Margin="10,134,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Почтовый индекс" HorizontalAlignment="Left" Margin="10,174,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Адрес" HorizontalAlignment="Left" Margin="10,214,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Компания" HorizontalAlignment="Left" Margin="10,256,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Профессия" HorizontalAlignment="Left" Margin="10,298,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Телефон" HorizontalAlignment="Left" Margin="10,339,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Почта" HorizontalAlignment="Left" Margin="10,380,0,0" VerticalAlignment="Top" Width="123"/>

<Label Content="Фото" HorizontalAlignment="Left" Margin="10,421,0,0" VerticalAlignment="Top" Width="123"/>

<TextBox Text="{Binding name}" HorizontalAlignment="Left" Height="36" Margin="138,10,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding middlename}" HorizontalAlignment="Left" Height="36" Margin="138,51,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding passportserial}" HorizontalAlignment="Left" Height="36" Margin="138,92,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding email}" HorizontalAlignment="Left" Height="36" Margin="138,379,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding phone}" HorizontalAlignment="Left" Height="36" Margin="138,338,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding jobname}" HorizontalAlignment="Left" Height="36" Margin="138,297,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding company}" HorizontalAlignment="Left" Height="36" Margin="138,256,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding address}" HorizontalAlignment="Left" Height="36" Margin="138,215,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding postcode}" HorizontalAlignment="Left" Height="36" Margin="138,174,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding passportnumber}" HorizontalAlignment="Left" Height="36" Margin="138,133,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

<TextBox Text="{Binding photo}" HorizontalAlignment="Left" Height="36" Margin="138,420,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="344"/>

</Grid>

</Page>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace GiBdd\_Zhiganov\_Vlad.Pages

{

/// <summary>

/// Логика взаимодействия для AddUserPage.xaml

/// </summary>

public partial class AddUserPage : Page

{

private Drivers \_currentUser = new Drivers();

public AddUserPage(Drivers selectedUser)

{

InitializeComponent();

if (selectedUser != null)

\_currentUser = selectedUser;

DataContext = \_currentUser;

}

private void ButtonCancel\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new UserPage());

}

private void ButtonSave\_Click(object sender, RoutedEventArgs e)

{

StringBuilder errors = new StringBuilder();

if (string.IsNullOrWhiteSpace(\_currentUser.name))

errors.AppendLine("Укажите имя и фамилию!");

if (string.IsNullOrWhiteSpace(\_currentUser.middlename))

errors.AppendLine("Укажите отчество");

if (string.IsNullOrWhiteSpace(\_currentUser.passportserial))

errors.AppendLine("Укажите серию паспорта");

if (string.IsNullOrWhiteSpace(\_currentUser.passportnumber))

errors.AppendLine("Укажите номер паспорта!");

if (string.IsNullOrWhiteSpace(\_currentUser.postcode))

errors.AppendLine("Укажите индекс!");

if (string.IsNullOrWhiteSpace(\_currentUser.address))

errors.AppendLine("Укажите адрес!");

if (string.IsNullOrWhiteSpace(\_currentUser.company))

errors.AppendLine("Укажите компанию!");

if (string.IsNullOrWhiteSpace(\_currentUser.jobname))

errors.AppendLine("Укажите профессию");

if (string.IsNullOrWhiteSpace(\_currentUser.phone))

errors.AppendLine("Укажите телефон!");

if (string.IsNullOrWhiteSpace(\_currentUser.email))

errors.AppendLine("Укажите почту");

if (string.IsNullOrWhiteSpace(\_currentUser.photo))

errors.AppendLine("Укажите фото!");

if (errors.Length > 0)

{

MessageBox.Show(errors.ToString());

return;

}

if (\_currentUser.id == 0)

Entities1.GetContext().Drivers.Add(\_currentUser);

try

{

Entities1.GetContext().SaveChanges();

MessageBox.Show("Данные успешно сохранены!");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

NavigationService.Navigate(new UserPage());

}

}

}



Рисунок 4. MessageBox

<Page x:Class="GiBdd\_ Zhiganov\_Vlad.Pages.Page1"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:GiBdd\_ Zhiganov\_Vlad.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Page1">

<Grid Background="#FFB5D5CA">

<Label Content="Повторите" HorizontalAlignment="Left" Margin="93,64,0,0" VerticalAlignment="Top" Height="48" Width="248" FontSize="16"/>

<Button Content="Показать" HorizontalAlignment="Left" Margin="93,112,0,0" VerticalAlignment="Top" Width="248" Click="Button\_Click\_1"/>

<TextBox x:Name="Word" IsReadOnly="True" HorizontalAlignment="Left" Margin="93,175,0,0" VerticalAlignment="Top" Width="248" Height="59" FontSize="24"/>

<TextBox x:Name="Repeat" HorizontalAlignment="Left" Height="67" Margin="93,239,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="248"/>

<Button Content="Сохранить" HorizontalAlignment="Left" Margin="93,311,0,0" VerticalAlignment="Top" Width="248" Click="Button\_Click"/>

</Grid>

</Page>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace GiBdd\_Zhiganov\_Vlad.Pages

{

/// <summary>

/// </summary>

public partial class Page1 : Page

{

public Page1()

{

InitializeComponent();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new UserPage());

}

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

string s = "";

Random r = new Random();

for (int i = 0; i < 5; i++)

s += (char)(r.Next(1072, 1104));

Word.Text = s;

}

}

}

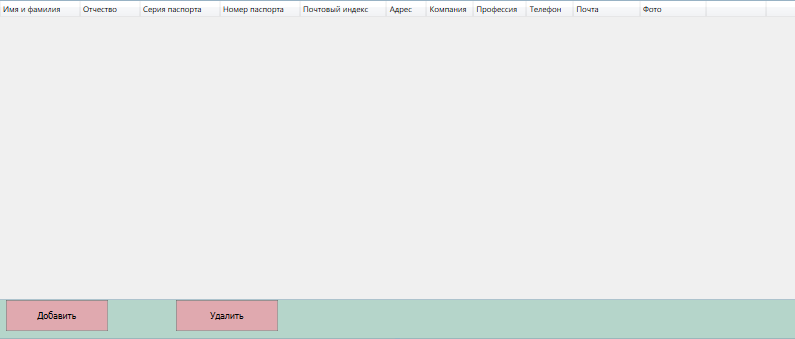


Рисунок 5. Стриница пользователя

<Page x:Class="GiBdd\_ Zhiganov\_Vlad.Pages.UserPage"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:GiBdd\_Zhiganov\_Vlad.Pages"

mc:Ignorable="d"

Title="UserPage" Height="508.274" Width="1197.123">

<Grid Background="#FFB5D5CA" Margin="0,0,-97,0" >

<Grid.ColumnDefinitions>

<ColumnDefinition/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="199\*"/>

<RowDefinition Height="26\*"/>

</Grid.RowDefinitions>

<DataGrid Grid.Row="0" Name="DataGridUser" IsVisibleChanged="DataGridUser\_IsVisibleChanged" AutoGenerateColumns="False" IsReadOnly="True">

<DataGrid.Columns>

<DataGridTextColumn Header="Имя и фамилия" Binding="{Binding name}" Width="120"/>

<DataGridTextColumn Header="Oтчество" Binding="{Binding middlename}" Width="90"/>

<DataGridTextColumn Header="Серия паспорта" Binding="{Binding passportserial}" Width="120"/>

<DataGridTextColumn Header="Номер паспорта" Binding="{Binding passportnumber}" Width="120"/>

<DataGridTextColumn Header="Почтовый индекс" Binding="{Binding postcode}" Width="130"/>

<DataGridTextColumn Header="Адрес" Binding="{Binding address}" Width="60"/>

<DataGridTextColumn Header="Компания" Binding="{Binding company}" Width="70"/>

<DataGridTextColumn Header="Профессия" Binding="{Binding jobname}" Width="80"/>

<DataGridTextColumn Header="Телефон" Binding="{Binding phone}" Width="70"/>

<DataGridTextColumn Header="Почта" Binding="{Binding email}" Width="100"/>

<DataGridTextColumn Header="Фото" Binding="{Binding photo}" Width="100"/>

<DataGridTemplateColumn Width="90">

<DataGridTemplateColumn.CellTemplate>

<DataTemplate>

<Button Name="ButtonEdit" Click="ButtonEdit\_Click" Content="Редактировать"></Button>

</DataTemplate>

</DataGridTemplateColumn.CellTemplate>

</DataGridTemplateColumn>

<DataGridTemplateColumn Width="90">

<DataGridTemplateColumn.CellTemplate>

<DataTemplate>

<Button Name="card" Click="card\_Click\_1" Content="Карточка"></Button>

</DataTemplate>

</DataGridTemplateColumn.CellTemplate>

</DataGridTemplateColumn>

</DataGrid.Columns>

</DataGrid>

<Button Grid.Row="1" Name="ButtonAdd" ToolTip="Добавить" Content="Добавить" HorizontalAlignment="Left" Click="ButtonAdd\_Click" Height="Auto" Width="153" Margin="10,0,0,12"/>

<Button Grid.Row="1" Name="ButtonDel" ToolTip="Удалить" Content="Удалить" HorizontalAlignment="Right" Click="ButtonDel\_Click" Height="Auto" Width="153" Margin="0,0,876,12"/>

</Grid>

</Page>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.InteropServices;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Interop;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Windows.Threading;

namespace GiBdd\_Zhiganov\_Vlad.Pages

{

/// <summary>

/// </summary>

public partial class UserPage : Page

{

DispatcherTimer \_timer;

public UserPage()

{

InitializeComponent();

\_timer = new DispatcherTimer();

\_timer.Interval = new TimeSpan(0, 0, 0, 3);

\_timer.Tick += IdleTick;

\_timer.Start();

}

private void IdleTick(object sender, EventArgs e)

{

var idle = GetIdle();

if (idle.Seconds >= 10)

{

MessageBox.Show("Zzzz...");

}

}

TimeSpan GetIdle()

{

var lastInputInfo = new LASTINPUTINFO();

lastInputInfo.cbSize = (uint)Marshal.SizeOf(lastInputInfo);

GetLastInputInfo(ref lastInputInfo);

var lastInput = DateTime.Now.AddMilliseconds(

-(Environment.TickCount - lastInputInfo.dwTime));

return DateTime.Now - lastInput;

}

[DllImport("User32.dll")]

private static extern bool GetLastInputInfo(ref LASTINPUTINFO plii);

[StructLayout(LayoutKind.Sequential)]

internal struct LASTINPUTINFO

{

public uint cbSize;

public uint dwTime;

}

private void ButtonEdit\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.AddUserPage((sender as Button).DataContext as Drivers));

}

private void ButtonAdd\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new AddUserPage(null));

}

private void ButtonDel\_Click(object sender, RoutedEventArgs e)

{

var usersForRemoving = DataGridUser.SelectedItems.Cast<Drivers>().ToList();

if (MessageBox.Show($"Вы точно хотите удалить записи в количестве {usersForRemoving.Count()} элементов?", "Внимание",

MessageBoxButton.YesNo, MessageBoxImage.Question) == MessageBoxResult.Yes)

try

{

Entities1.GetContext().Drivers.RemoveRange(usersForRemoving);

Entities1.GetContext().SaveChanges();

MessageBox.Show("Данные успешно удалены!");

DataGridUser.ItemsSource = Entities1.GetContext().Drivers.ToList();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

}

private void DataGridUser\_IsVisibleChanged(object sender, DependencyPropertyChangedEventArgs e)

{

if (Visibility == Visibility.Visible)

{

Entities1.GetContext().ChangeTracker.Entries().ToList().ForEach(x => x.Reload());

DataGridUser.ItemsSource = Entities1.GetContext().Drivers.ToList();

}

}

private void card\_Click\_1(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new card(null));

}

}

}

Изображение выглядит как Прямоугольник

Автоматически созданное описание

Рисунок 6. Страница данных

<Window x:Class="GiBdd\_ Zhiganov\_Vlad.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:GiBdd\_ Zhiganov\_Vlad"

mc:Ignorable="d"

Title="MainWindow" Height="573.944" Width="1197.123">

<Grid Background="#FFB5D5CA">

<Grid.RowDefinitions>

<RowDefinition Height="25.751"/>

<RowDefinition Height="517"/>

<RowDefinition Height="0"/>

</Grid.RowDefinitions>

<Frame x:Name="MainFrame" NavigationUIVisibility="Hidden" Navigated="MainFrame\_Navigated" Source="Pages/AuthPage.xaml" Grid.RowSpan="3" Margin="0,0,0,-10" />

<Label x:Name="Data" HorizontalAlignment="Left" Margin="623,15,0,-50" VerticalAlignment="Top" Width="16" Grid.Row="2"/>

<Button x:Name="ButtonBack" Content="Назад" HorizontalAlignment="Left" Margin="0" VerticalAlignment="Top" Width="23" Click="ButtonBack\_Click" Height="25" Background="#FFFFFCD6"/>

</Grid>

</Window>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.InteropServices;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Windows.Threading;

namespace GiBdd\_Zhiganov\_Vlad

{

public partial class MainWindow : Window

{

DispatcherTimer \_timer;

public MainWindow()

{

InitializeComponent();

\_timer = new DispatcherTimer();

\_timer.Interval = new TimeSpan(0, 0, 0, 3);

\_timer.Tick += IdleTick;

\_timer.Start();

}

private void IdleTick(object sender, EventArgs e)

{

var idle = GetIdle();

if (idle.Seconds >= 10)

{

MessageBox.Show("Zzzz...");

}

}

TimeSpan GetIdle()

{

var lastInputInfo = new LASTINPUTINFO();

lastInputInfo.cbSize = (uint)Marshal.SizeOf(lastInputInfo);

GetLastInputInfo(ref lastInputInfo);

var lastInput = DateTime.Now.AddMilliseconds(

-(Environment.TickCount - lastInputInfo.dwTime));

return DateTime.Now - lastInput;

}

[DllImport("User32.dll")]

private static extern bool GetLastInputInfo(ref LASTINPUTINFO plii);

[StructLayout(LayoutKind.Sequential)]

internal struct LASTINPUTINFO

{

public uint cbSize;

public uint dwTime;

}

private void MainFrame\_Navigated(object sender, NavigationEventArgs e)

{

}

private void ButtonBack\_Click(object sender, RoutedEventArgs e)

{

if (MainFrame.CanGoBack) MainFrame.GoBack();

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ №1**

# **Тема: «Дизайн на HTML и CSS»**

Студенты: Жиганов В. О.

Группа: 4ИСИП-519

Преподаватель: Пташкин О. Г.

Дата: 14.03.2023

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. Шапка сайта



Рисунок 2. Срок работы на рынке

Изображение выглядит как текст, снимок экрана, в помещении

Автоматически созданное описание

Рисунок 3. Реклама акции

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4. Наши услуги

Изображение выглядит как карта

Автоматически созданное описание

Рисунок 5. Сайт «Недвижимость»

Код:

Style.css:

li

{

    display: inline-block;

    list-style: none;

  }

.toptxt

{

    font-size: 14px;

    margin-left: 20px;

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-style: normal;

    font-weight: 400;

    display: flex;

    align-items: center;

    text-align: center;

    color: #000000;

}

.mainheader

{

    font-size: 60px;

    margin-top: 150px;

    margin-left: 150px;

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-weight: 600;

    font-style: normal;

}

.underheader

 {

    margin-left: 150px;

    color: #2B2B2B;

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-weight: 400;

    font-style: normal;

    font-size: 16px;

}

.getplanandcost

{

    background-color: #207FF1;

    color: white;

    margin-left: 150px;

    border-radius: 15px;

    height: 35px;

    border-color: #207FF1;

    border-bottom: #207FF1;

    border-right: #207FF1;

}

.img

{

    margin-top: 80px;

    height: 750px;

    width: 1920px;

}

.zaslygi

{

    margin-left: 150px;

width: 230px;

height: 53px;

left: 450px;

top: 2883px;

font-family: 'SF UI Text';

font-style: normal;

font-weight: 600;

font-size: 32px;

line-height: 53px;

}

.card\_1

{

    width: 332.5px;

    height: 100px;

    background: #ffffff;

    margin-left: 50px;

    border-radius: 15px;

    text-align: center

}

.card\_2

{

    width: 332.5px;

    height: 100px;

    background: #ffffff;

    margin-left: 50px;

    border-radius: 15px;

    text-align: center

}

.card\_3

{

    width: 332.5px;

    height: 100px;

    background: #ffffff;

    margin-left: 50px;

    border-radius: 15px;

    text-align: center

}

.card\_4

{

    width: 332.5px;

    height: 100px;

    background: #ffffff;

    margin-left: 50px;

    border-radius: 15px;

    text-align: center

}

.dop

{

width: 272px;

height: 36px;

padding-left: 100px;

color: #767676;

}

.bignum

{

    font-size: 40px;

    color: #207FF1;

}

.vigoda

{

    font-family: 'SF UI Text';

font-style: normal;

font-weight: 500;

font-size: 23px;

line-height: 30px;

}

.podrobnee

{

padding: 6px 20px;

gap: 10px;

left: 295px;

top: 252px;

bottom: 100px;

background: #F1F6FC;

border-radius: 20px;

}

.data

{

font-size: 15px;

line-height: 23px;

color: #888888;

opacity: 0.9;

}

.getcall

{

    border-radius: 12px;

    background-color: white;

    color: #207FF1;

    border-color: #207FF1;

    height: 35px;

}

.card2

{

    background-color: white;

    width: 450px;

    height: 312;

    border-radius: 30px;

}

.mizapystili

{

    font-family: 'SF UI Text';

font-style: normal;

font-weight: 400;

font-size: 15px;

line-height: 23px;

/\* or 153% \*/

color: #767676;

opacity: 0.9;

}

.txtupimg

{

    position: relative;

    }

.circleangles

{

    border-radius: 15px;

    height: 230px;

    width: 160px;

}

.txt1

{

    top: 140px;

    left: 10px;

    font-family: 'Roboto';

    font-style: normal;

    font-weight: 700;

    font-size: 20px;

    line-height: 22px;

    color: #FFFFFF;

}

.imgwrap

{

    position: relative;

}

.imgwrap h2

{

    position: absolute;

}

.imgwrap p

{

    position: absolute;

}

.imgwrap button

 {

    position: absolute;

}

.onimgtext

{

    margin-top: -550px;

    margin-left: 200px;

    color: white;

}

.onimgtext2

{

    margin-top: -470px;

    margin-left: 200px;

    color: white;

}

.phonebox

{

    background-color: white;

    border-radius: 10px;

    position: absolute;

    width: 200px;

    border: 1px solid black;

    margin-top: -250px;

    margin-left: 200px;

}

.btnunderphonebox

 {

    margin-top: -200px;

    margin-left: 200px;

    height: 35px;

    width: 200px;

    border-radius: 10px;

    border-color: #000000;

    color: #000000;

 }

.logo

{

    margin-left: 100px;

    width: 40px;

    height: 40px;

    vertical-align: middle;

}

.otdelprodaj

{

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-style: normal;

    font-weight: 800;

    font-size: 18px;

    line-height: 24px;

    color: #000000;

    vertical-align: middle;

}

.phoneicon

 {

    vertical-align: middle;

    margin-left: 20px;

}

.AHotdel

{

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-style: normal;

    font-weight: 600;

    font-size: 32px;

    color: #000000;

}

.AHotdel2

{

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    font-style: normal;

    font-weight: 400;

    font-size: 18px;

    color: #2B2B2B;

    margin-left: 100px;

}

.map

{

width: 700px;

height: 300px;

border-radius: 40px;

margin-left: 200px;

}

.backcolbutton

{

    background-color: #207FF1;

    color: white;

    border-radius: 15px;

    height: 35px;

    margin-left: 50px;

    border-color: #207FF1;

    border-bottom: #207FF1;

    border-right: #207FF1;

}

.city

{

    width: 700px;

    height: 300px;

    border-radius: 40px;

    margin-left: 50px;

}

.contactname

{

    width: 700px;

    height: 300px;

    border-radius: 40px;

    margin-left: 50px;

}

.phone

{

    width: 700px;

    height: 300px;

    border-radius: 40px;

    margin-left: 50px;

}

.backcolimg

{

    border-radius: 15px;

    height: 20px;

    margin-left: 50px;

}

.backcolimg\_2

{

    border-radius: 15px;

    height: 20px;

    margin-left: 5x;

}

.backcolimg\_3

{

    border-radius: 5px;

    height: 18px;

    margin-left: 5x;

}

Index.html:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <link rel="stylesheet" href="style.css">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <ul>

        <li><img class="logo" src="images/logotip.png"></li>

        <li class="otdelprodaj"><p>ОТДЕЛ ПРОДАЖ</p></li>

        <li><p class="toptxt" style="margin-left: 50px;">Каталог объектов</p></li>

        <li><p class="toptxt">Блог</p></li>

        <li><p class="toptxt">Команда</p></li>

        <li><p class="toptxt">FAQ</p></li>

        <li><p class="toptxt">Контакты</p></li>

        <li><img class="phoneicon" src="images/phone.png"></li>

        <li><p>+7 (495) 659-99-88</p></li>

        <li><button class="getcall">Заказать звонок</button></li>

    </ul>

    <h1 class="mainheader">Исскусство обращения<br>с недвижимостью</h1>

    <p class="underheader">Недвижимость - это то, чем мы живём и то, что<br>заставляет нас двигаться вперёд. Отвечаем за<br>результат с 2015 года</p>

    <button class="getplanandcost">Получить планировки и цены</button>

    <br>

    <img class="img" src="images/img1.png">

    <div style="background-color: #F1F6FC;">

    <ul style="margin-left: 100px;">

        <li><h2 class="AHotdel">АН "Отдел продаж" - 15 лет<br>на рынке сочи</h2></li>

        <li><p class="AHotdel2">Более 15 лет наши специалисты в области недвижимости<br>и права помогают покупать квартиры максимально<br>безопасно дёшево</p></li>

    </ul>

    <ul style="margin-left: 100px;">

        <li><div class="card\_1">

            <p><span class="num">16 домов</span></p>

            <p>Полностью заселены<br>довольными клиентами</span></p>

        </div></li>

        <li><div class="card\_2">

            <p><span class="num">3000 квартир</span></p>

            <p>Полностью заселены<br>довольными клиентами</span></p>

        </div></li>

        <li><div class="card\_3">

            <p><span class="num">2000 человек</span></p>

            <p>Полностью заселены<br>довольными клиентами</span></p>

        </div></li>

        <li><div class="card\_4">

            <p><span class="num">350+ сделок</span></p>

            <p>Полностью заселены<br>довольными клиентами</span></p>

        </div></li>

    </ul>

    </div>

    <div class="imgwrap">

    <img src="images/img2.png">

    <h2 class="onimgtext">Получите выгодное<br>предложение!</h2>

    <p class="onimgtext2">Ещё год назад цены на недвижимость<br>были ниже на 34%.<br>

        Лучшее время для покупки квартиры -<br>"прошлой зимой".</p>

    <div class="phonebox">

        <p>+7 (\_\_\_) \_\_\_ - \_\_ - \_\_</p>

    </div>

    <button class="btnunderphonebox">Оставить заявку</button>

    </div>

    <br>

    <br>

    <br>

    <h2><span class="zaslygi"> Наши заслуги</span></h2>

    <ul>

        <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button  class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

            <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

            <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

    </ul>

    <ul>

        <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

            <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

            <li><div class="card2">

                <p><span class="vigoda">Выгодная ставка от 6,5% на<br>покупку коммерческой<br>недвижимости</span></p>

                <p><span class="mizapystili">Мы запустили программу субсидированной<br>ипотеки для Сочинских предпринимателей<br>малого и среднего бизнеса.</span></p>

                <ul>

                    <li><p class="data">24.05.2022</p></li>

                    <li><button class="podrobnee">Подробнее</button></li>

                </ul>

            </div></li>

    </ul>

    <ul>

        <li><h2><span class="contactname">Контакты</span></h2>

        <p><span class="phone">+7 (495) 659-99-88</span></p>

        <p><span class="city">г. Сочи,ул. Конституции СССР 4,2 этаж, офис 224</span></p>

        <button class="backcolbutton">Обратный звонок</button>

        <br>

        <div><img class="backcolimg" src="images/WhatsUp.png"><img class="backcolimg\_2"src="images/phone.png"><img class="backcolimg\_3" src="images/eMail.png"></div>

        </li>

        <li><img  class="map" src="images/map.png"></li>

    </ul>

</body>

</html>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ №1**

# **Тема: «HTML и CSS на Jupiter»**

Студент: Жиганов В. О.

Группа: 4ИСИП-519

Преподаватель: Сибирев И. В.

Дата: 14.03.2023

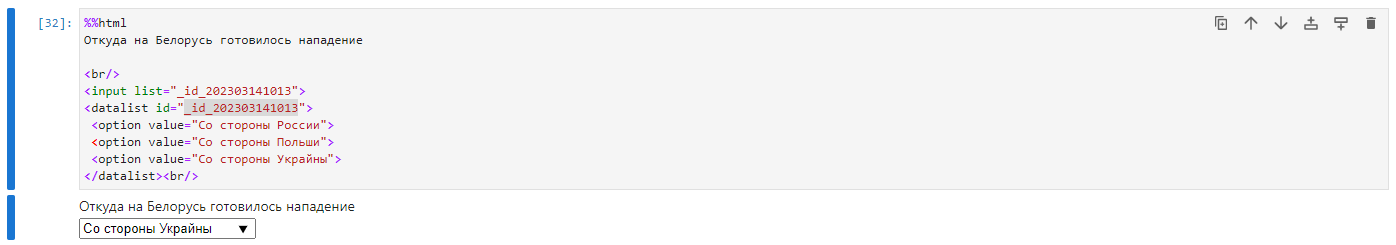


Рисунок 1: Выбор вариант на панели

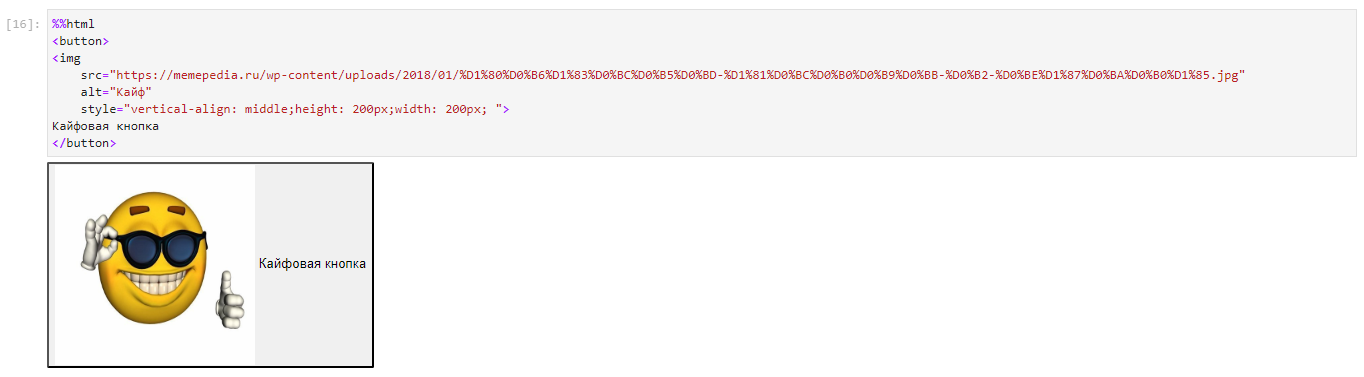


Рисунок 2: Кнопка с jpg картинкой